DYNAMIC PROXY RECONFIGURATION SYSTEM AND METHOD TO SUPPORT SHARING OF EXTRA CAPACITY

Field of the Invention

The invention generally relates to a system and method for dynamically reconfiguring proxy server networks so that they may share extra capacity.

Background

Network systems, such as the internet and, more specifically, the World Wide Web ("WWW"), utilize servers to process requests for information. However, with increased popularity, these network systems are gradually becoming more overloaded since the number of requests for information has been sharply increasing. When a server becomes overloaded, it may be unable to receive new requests, may be slow to process the requests it has already received, and may yield server errors.

On the WWW, such overloading can be extremely detrimental. As web browsers have become the primary interface for access to many network and server services, more businesses have begun using the WWW to market their products or for advertising purposes. To these businesses,

prompt and efficient access to their sites is extremely desirable, as it is well known that WWW users tend to avoid sites which load slowly or yield server errors.

However, when a popular web page is hosted by a single computer or server, the internet traffic to the computer can be overwhelming. To address this situation, a number of server-based solutions have been proposed and/or implemented to attempt to ensure that Internet services remain available, scalable and well-balanced. One of the most popular solutions has been the use of load balancing, which involves distributing requests among servers (e.g., different servers on a WWW site) in order to ensure that any one server does not become unduly burdened.

One conventional load balancing technique involves the use of a domain name server (hereinafter "DNS"). This device is responsible for resolving uniform resource locators or "URLs" (e.g., "www.foo.com") to specific IP addresses (e.g., 111.222.111.222). In this regard, a Web site having several servers may operate under a single URL, although each server is assigned a different IP address. It is up to the DNS to determine which server to route a web user to when a request is made. For example, a round-

robin DNS performs load balancing by routing requests to these servers in sequential rotation based on their IP addresses.

When a web site has several servers operating under the same URL, those extra servers are often called "proxy" or "mirror" servers. The proxy server stores exactly the same web site information found in the originating server. Thus, when a user makes a request to visit a website that uses a proxy server, the user will see the same exact website whether or not the user is visiting the proxy server or the original server.

As the need for proxy services has increased, some companies have begun to operate their own proxy networks for the purpose of delivering content for certain subscriber WWW sites. However, the agreements these proxy network providers have with the subscriber WWW sites is generally long-term in nature. Thus, the proxy networks are generally configured for the maximum expected traffic at the subscribed WWW sites and there is often unused capacity on the proxy network being wasted. What is needed is a method to let the proxy network dynamically sell the extra capacity so that it is not wasted.

Summary

The system and method is directed to dynamic proxy reconfiguration implemented in interconnected network servers. In particular, the invention is directed to a computer-executable program for use in proxy network servers which enables each proxy server to dynamically sell its unused capacity to other web sites for specific periods of time. The invention has particular utility in connection with World Wide Web servers and proxy servers, but can be used with other servers where proxy servers may be present, such as CORBA servers, ORB servers, FTP servers, SMTP servers, and Java servers. The system and method may be used to dynamically sell extra capacity to other websites to make additional profit.

In a preferred embodiment, a proxy server network monitors its servers to determine whether any unused capacity exists. If any unused capacity exists, the proxy server can sell an estimated or set amount of unused capacity for a set amount of time through various market mechanisms to web site server operators.

Once a purchaser has been identified, a controller program either on the domain name server of the proxy

network or on a separate server connected to the proxy server network ensures that the proxy network's domain name server receives information on the purchaser website. This information includes the name to address map of the purchaser web site network and the content of those websites which will be stored for the purchased period of time on the proxy servers.

After this information is stored on the domain name server of the proxy server network, the domain name server of the proxy server network can begin mapping a fraction of the overall mapping requests to the proxy servers. The overall fraction of requests mapped by the domain name server will depend on the initial agreement between the proxy network and the purchaser. For example, if the unused proxy capacity was determined based on an estimate of extra capacity available, the proxy network might service the purchaser website's mapping requests using its best efforts for the time its agreed to provide proxy server capacity to the purchaser. In such a case, the final bill due the proxy server network will be based on the purchaser website's actual usage of the proxy server capacity.

The remaining fraction of mapping requests which the proxy server network does not handle are routed back to the purchaser website's servers for mapping. However, if the purchaser supplied the controller program with an assignment algorithm, the domain name server of the proxy server network will route the remaining mapping requests to servers in the purchaser website's network.

Brief Description of the Drawings

Figure 1 is a hardware diagram illustrating the data flow of the preferred embodiment for the system and method to let a proxy network dynamically sell unused capacity.

Figure 2 is a hardware diagram illustrating the data flow of an alternative embodiment for the system and method to let a proxy network dynamically sell unused capacity.

Figure 3 is a block diagram illustrating a typical path taken by a user's request for a particular address on the internet, and the path taken in receiving that address.

Figures 4-8 are flowcharts illustrating embodiments of the system and method to let a proxy network dynamically sell unused capacity.

Detailed Description of the Preferred Embodiments

The system and method is directed to reconfiguring proxy network servers so that proxy networks can dynamically sell unused capacity to other networks for specific periods of time. This unused capacity can be used for delivering content normally found on the purchaser's network, so that the overall load on the purchaser network is reduced. Although the system and method will be described in the context of the WWW, and more specifically the content of WWW servers, it is not limited to use in this context. Rather, the system and method can be used in a variety of different types of networking systems with a variety of different servers. For example, the system and method can be used in intranets and local area networks, and with CORBA servers, ORB servers, FTP servers, SMTP servers, and Java servers, to name a few.

Fig. 1 is a basic hardware diagram of a networking system capable of carrying out the processing in accordance to one embodiment of the system and method. More specifically, Fig. 1 depicts the topology of a website network and proxy network and how they connect to the internet and to each other. Fig. 1 shows a proxy network's

connections to the internet consisting of a proxy network server cluster 5, a packet filter 4, a local domain name server of the proxy server network which contains a controller program 3, and a router for the proxy network 2 to connect to the internet 1. A similar set of hardware components exists for a purchaser web site server cluster 9 to connect to the internet 1. A brief description of this hardware is provided below.

Router 2 receives requests for information stored on the proxy network server cluster 5 from a remote location via the internet 1. Router 2 routes these requests, which typically comprise URL's, to the local domain name server of the proxy server network 3. The local domain name server of the proxy network 3 receives a URL from router 2 and resolves the domain name in the URL to a specific IP address in proxy server cluster 5.

Router 6 and local domain name server 7 perform the same tasks as Router 2 and domain name server 3, except that they are routing requests to a purchaser web site server cluster 9.

Packet filters 4 and 8 are generally found in most networks and serve as firewalls for the internal networks

consisting of proxy network server cluster 5 and purchaser web site server cluster 9 respectively. All transactions into and out of an internal network are handled by the network's packet filter. Packet filters determine which services of the internal network may be accessed from the internet 1, which clients are permitted access to those services, and which outside services may be accessed by anyone on the internal network. Thus, packet filters 4 and 8 analyze data packets passing through them and filter those packets according to the settings on each network, restricting access where necessary and allowing access where appropriate.

The proxy network server cluster 5 and the purchaser web site server cluster 9 are both internal networks which are typically comprised of multiple servers. Sometimes these servers are all connected through a mainframe (or similar computer). The servers that make up each server cluster 5 and 9 are used to store files, such as website files, so that users may later access and view the files.

Fig. 2 illustrates an alternate embodiment to the hardware structure depicted in Fig. 1. In Fig. 2, instead of storing the controller program on the local domain name

server of the proxy network 3, the controller program is located on its own server 10 which is connected to the local domain name server 11.

Fig. 3 is a hardware diagram depicting the general path taken by a user's request for a particular address on the internet, and the path taken in receiving that address used in the prior art. A user 50 using a web browser requests a web site address using a URL. The URL is then sent to a local domain name server 51 on the user's 50 own local network. Domain name servers are responsible for resolving uniform resource locators or "URLs" (e.g., "www.foo.com") to specific internet or internet provider ("IP") addresses (e.g., 111.222.111.222). If the user 50 is requesting an address on the local network, the local domain name server 51 will have the corresponding internet address and will relay the internet address back to the user 50. The web browser of the user 50 will then take the user 50 directly to the requested site.

Otherwise, if the user 50 is not requesting a URL which corresponds to an IP address on the local network, the local domain name server 51 will not have corresponding IP address and the local domain name server 51 will have to

contact a root domain name server 52 to get the information. If the root domain name server 52 does not have the IP address corresponding to the URL submitted by the user 50, then various minor domain name servers 53 and 54 will have to be contacted. If the minor domain name servers 53 and 54 don't have the internet address, the local domain name server 55 of the network 56 the user is contacting will be contacted to provide the IP address. However, often the minor domain name servers 53 and 54 will have the IP address to the URL requested by the user, and the local domain name server 55 of the network that the user is contacting will not have to be contacted.

Fig. 4 illustrates the initial process steps of the present invention for dynamically reconfiguring a proxy network to sell extra capacity to other networks, specifically WWW server networks. To begin, in steps 100 and 101 a determination must be made as to whether any unused proxy server capacity actually exists for a period of time which could be marketed to a web site operator. This determination can be made by a proxy network operator or by the controller program monitoring the proxy server network. The controller program is stored either on its

own server 10 connected to the proxy server network, or on the local domain name server of the proxy server network 3.

Once it is determined that some unused proxy server capacity is available, step 102 follows where the unused proxy server capacity is marketed through various mechanisms known to persons skilled in the art. Some of the various ways in which the unused proxy server capacity can be marketed includes, but is not limited to, online auctions on the internet or on real-time continuous markets which are accessible via the internet. The unused proxy server capacity can also be sold either as an estimate (i.e., the proxy server network will use its best efforts to provide the capacity being sold and will possibly even supply additional capacity if it becomes available at a predetermined rate) or for a specific amount. In either case, the unused proxy server capacity will be sold to a purchaser for a limited, set time (i.e., the proxy server network will accept purchaser website's requests for four hours or for four hours on a daily basis). Payment for the purchase of proxy server capacity can be made through various mechanisms known to persons skilled in the art.

For example, a credit card could be used or money could be wired from a specific account.

If some unused proxy server capacity is sold, step 103 then follows where the controller on the proxy server network is notified about the sale. Purchaser information is then sent to the controller in step 200 in Fig. 5. Such information includes, but is not restricted to, the purchaser's billing information such as credit card information, billing address, etc..

In step 201, the controller program ensures that the local domain name server of the proxy network is the primary domain name server, which is the only domain name server that can assign names to the proxy servers. As illustrated in Fig. 3, a user 50 who requests a website using a URL must have the URL mapped to a numerical IP address before accessing the actual website. However, often the user 50 does not have to contact the local domain name server of the network to get the IP address which corresponds to the URL the user 50 is requesting. In such cases, other minor domain name servers 53 and 54, which are outside the network, are able to provide the user 50 with the requested IP address. Step 201 ensures that local

domain name server of the proxy network 55 will serve all naming requests. Therefore, the root domain name server 52 and minor domain name servers 53 and 54 will not be able to provide the user 50 with any name to address translations for the proxy server network. In this fashion, only the local domain name server 55 will have to be updated when proxy server network dynamically provides unused capacity.

In step 202, the primary domain name server receives the name to address map of the purchaser web site and routes copies of the purchaser's website content to servers on the proxy network. Therefore, the primary domain name server 3 handles all name to address translation requests for the purchaser website for the time that the purchaser has paid to use the proxy server capacity. The name to address map of the purchaser website can be obtained by the primary domain name server from the purchaser website's local domain name server. In addition it could be sent to the primary domain name server by a purchaser website operator along with an assignment algorithm or other mapping information the purchaser would like the proxy server network to know. The controller program would

ensure that the information is received and handled appropriately.

In step 203, the controller program determines how to handle the mapping requests to the purchaser's website by examining whether or not the unused proxy capacity was purchased based on an estimate of usage. For example, if the original sale of the proxy server capacity was based on an estimate of unused capacity available, or if purchaser just wanted to purchase whatever unused proxy capacity existed, the controller program will have to initiate a steps 204 and 205 to route the mapping requests for the purchaser.

In step 204, the primary domain name server routes a fraction of the overall mapping requests for the purchaser website to servers in the proxy network based on the amount of unused proxy capacity available. In step 205, the primary domain name server monitors the load levels on the proxy servers to adjust the fraction of mapping requests for the purchaser website routed to proxy servers based on the amount of unused capacity available at any given time. This ensures that the proxy server network is never overburdened by the number of requests to the purchaser

website. Thus, the dynamic sale of proxy capacity to other networks never interferes with the other operations of the proxy server network.

In step 206, the primary domain name server determines whether there are any mapping requests that cannot be routed to the proxy server due to a lack of proxy server capacity. If there are some, the primary domain name server next checks to see if the purchaser of the proxy capacity provided an assignment algorithm for handling these requests in step 300 in Fig. 6. If an assignment algorithm was provided, the primary domain name server routes all the mapping requests that the proxy network cannot handle to servers in the purchaser website's network in step 301. Otherwise, if an assignment algorithm was not provided, step 302 ensures that those mapping requests are returned to the domain name server of the purchaser website's network.

At this point, the controller program determines in step 500 in Fig. 8 whether the purchaser's period of time for using capacity on the proxy server network has expired. If it hasn't, the primary domain name server continues to map requests for the purchaser website. Otherwise, in step

501, a billing cycle is initiated. At this point, the controller program will determine whether or not to bill the purchaser a set amount or for the actual usage of the proxy network. If set amount of unused proxy capacity was originally purchased in step 102, the purchaser is billed the agreed upon amount as demonstrated in step 504. Otherwise, the purchaser is billed in step 503 for the overall actual usage of the proxy server capacity as determined by the controller program which monitored the purchaser's use of proxy capacity. At this point the controller program ends with regards to a particular purchaser, and begins step 100 to determine whether any unused proxy capacity is available.

Going back to step 203, if the purchaser did not buy proxy capacity based on an estimate, the primary domain name server will map a fraction of the received mapping requests to servers in the proxy network based on the actual proxy capacity purchased. Therefore, at no time will the amount of proxy capacity servicing mapping requests for the purchaser website's network be greater than the amount originally purchased. Additional mapping requests received by the primary domain name server which

are not mapped by the primary domain name server because it would require more proxy capacity than was purchased, will be handled in steps 402, 403, and 404 in the same fashion as steps 300, 302, and 303 discussed above.

Finally, an alternative embodiment exists as shown by Fig. 2 where the controller program is stored on its own controller server 10. This server is then responsible for handling all of the mapping requests much like the local domain name server 3.

The above description is presented to enable one of ordinary skill in the art to make and use the invention and is provided in the context of a patent application and its requirements. Various modifications to the preferred embodiments will be readily apparent to those skilled in the art and the generic principles herein may be applied to other embodiments. Thus, the present invention is not intended to be limited to the embodiment shown but is to be accorded the widest scope consistent with the principles and features described herein.